GlassBrowserAI Action Catalog TaskTree

Generated: 2026-01-09T18:00:04.792860Z

1. Task Tree Orchestrator

Default steps and statuses used to orchestrate playbook and task flows.

Default Steps

- observe
- evaluate
- decide
- verify
- execute
- monitor
- review

Statuses

- queued
- running
- completed
- failed
- skipped
- blocked

2. Action Catalog (services/actionCatalog.ts)

Each action has an ID, domain, safety gates, and default timeout.

Actions

- system.snapshot [system] - Fetch consolidated system state snapshot.
- action_flow.list [system] - List learned action flow recommendations.
- action_flow.run [system] - Run a learned action flow by intent key.
- session.bias.set [system] - Set the session bias for the trading team.
- symbol.scope.set [system] - Set the global symbol scope and default timeframes.
- symbol.scope.clear [system] - Clear the global symbol scope.
- chat.reply_mode.set [chat] - Set chat reply mode (single/team).
- chat.clear [chat] - Clear the chat conversation.
- chat.playbook.default.set [chat] - Set the default playbook used in Chat.
- chat.attachment.set [chat] - Set or clear the chat attachment.
- chat.snapshot.capture [chat] - Capture the active tab and attach to chat.

- chat.snapshot.send [chat] - Capture and send tab snapshots to chat.
- chat.send [chat] - Send a chat message through the active assistant.
- chat.auto_tab_vision.set [chat] - Toggle auto tab vision for chat.
- trade.execute [broker] - Execute a pending trade proposal.
- trade.reject [broker] - Reject a pending trade proposal.
- broker.action.execute [broker] - Execute a pending broker action.
- broker.action.reject [broker] - Reject a pending broker action.
- live.start [chat] - Start a live session (camera/screen/audio).
- live.stop [chat] - Stop the live session.
- live.mute [chat] - Mute live session audio capture.
- live.unmute [chat] - Unmute live session audio capture.
- post_trade_review.enable [chat] - Enable post-trade review automation.
- post_trade_review.disable [chat] - Disable post-trade review automation.
- post_trade_review.agent.set [chat] - Set the agent to run post-trade reviews.
- post_trade_review.run_last [chat] - Run post-trade review for the most recent closed trade.
- settings.openai.key.set [settings] - Store the OpenAI API key in secure storage.
- settings.openai.key.clear [settings] - Clear the OpenAI API key from secure storage.
- settings.gemini.key.set [settings] - Store the Gemini API key in secure storage.
- settings.gemini.key.clear [settings] - Clear the Gemini API key from secure storage.
- settings.model.text.set [settings] - Set OpenAI text model preference.
- settings.model.vision.set [settings] - Set OpenAI vision model preference.
- settings.model.vision_detail.set [settings] - Set OpenAI vision detail preference.
- settings.model.live.set [settings] - Set OpenAI live model preference.
- settings.vision_interval.active.set [settings] - Set active tab vision interval (ms).
- settings.vision_interval.watched.set [settings] - Set watched tab vision interval (ms).
- settings.chart_watch_interval.set [settings] - Set chart watch interval (ms).
- settings.auto_watch_tradingview.set [settings] - Toggle auto-watch TradingView tabs.
- settings.chat_context.mode.set [settings] - Set chat tab context mode.
- settings.chat_context.max_tabs.set [settings] - Set chat tab context max tabs.
- settings.chat_context.change_only.set [settings] - Toggle chat context change-only capture.
- settings.chat_context.roi.set [settings] - Set chat tab context ROI preset.
- settings.chat_context.redaction.set [settings] - Set chat tab context redaction preset.
- settings.broker.set_active [settings] - Set active broker adapter.
- ledger.flush [ledger] - Flush trade ledger to disk.
- ui.sidebar.setMode [ui] - Set sidebar mode (optionally open the sidebar).
- ui.sidebar.open [ui] - Open the sidebar.
- ui.sidebar.close [ui] - Close the sidebar.
- ui.sidebar.toggle [ui] - Toggle sidebar visibility.
- ui.panel.open [ui] - Open a specific sidebar panel.

- ui.modal.open [ui] - Open a modal dialog (settings).
- ui.modal.close [ui] - Close a modal dialog (settings).
- ui.command_palette.open [ui] - Open the command palette.
- ui.command_palette.close [ui] - Close the command palette.
- ui.command_palette.toggle [ui] - Toggle the command palette.
- ui.tab.open [ui] - Open a new browser tab with a URL.
- ui.tab.navigate [ui] - Navigate a tab to a URL.
- ui.tab.back [ui] - Navigate back in the active tab.
- ui.tab.forward [ui] - Navigate forward in the active tab.
- ui.tab.reload [ui] - Reload the active tab.
- ui.tab.switch [ui] - Switch to a browser tab by id, index, or url.
- ui.tab.close [ui] - Close a browser tab by id, index, or url.
- ui.tab.pin [ui] - Pin/unpin a tab for AI context.
- ui.tab.label.set [ui] - Set AI label for a browser tab.
- ui.tab.watch [ui] - Watch/unwatch a tab for background chart watch.
- ui.window.fullscreen.toggle [ui] - Toggle application fullscreen mode.
- tradelocker.view.set [ui] - Set active TradeLocker panel view.
- tradelocker.ticket.set [ui] - Update TradeLocker ticket inputs.
- tradelocker.orders.filters.set [ui] - Update TradeLocker orders filters.
- tradelocker.history.filters.set [ui] - Update TradeLocker history filters.
- tradelocker.close_panel.set [ui] - Update TradeLocker close position panel.
- backtester.config.set [backtest] - Update Backtester panel config inputs.
- backtester.params.set [backtest] - Update Backtester parameters for a strategy.
- backtester.optimizer.config.set [backtest] - Update Backtester optimizer config.
- backtester.batch.config.set [backtest] - Update Backtester batch config inputs.
- setups.library.filters.set [setup] - Update Setup Library filters.
- setups.compare.toggle [setup] - Toggle Setup Library compare selection.
- setups.playbook.draft.set [setup] - Update setup playbook draft fields.
- setups.playbook.save [setup] - Save the active playbook draft for a watcher.
- setups.playbook.reset [setup] - Reset the playbook draft for a watcher.
- setups.playbook.create.toggle [setup] - Toggle the new watcher playbook editor in Setups.
- setups.playbook.editor.toggle [setup] - Toggle setup playbook editor visibility for a watcher.
- setups.apply_to_backtester [setup] - Apply watcher params to the Backtester.
- setups.replay.run [setup] - Run a replay/backtest for a setup watcher.
- setups.debug.filters.set [setup] - Update setup debug replay filters.
- setups.create.form.set [setup] - Update setup watcher create form fields.
- dashboard.window.set [ui] - Set dashboard session window size.
- dashboard.smoothing.set [ui] - Toggle dashboard smoothing.
- dashboard.compare.set [ui] - Toggle dashboard compare mode.

- chart.snapshot [chart] - Capture native chart snapshot.
- chart.engine.snapshot [chart] - Fetch chart engine session snapshots.
- chart.watch.start [chart] - Start chart engine watch for symbol/timeframes.
- chart.watch.stop [chart] - Stop chart engine watch for symbol/timeframes.
- chart.watch.toggle [chart] - Toggle chart watch background capture.
- chart.watch.mode.set [chart] - Set chart watch mode.
- chart.watch.snooze [chart] - Snooze chart watch.
- chart.watch.clear_snooze [chart] - Clear chart watch snooze.
- chart.watch.lead_agent.set [chart] - Set lead agent for chart watch.
- chart.session.list [chart] - List chart sessions.
- chart.session.create [chart] - Create a chart session for a symbol.
- chart.session.create_from_tab [chart] - Create a chart session from a browser tab.
- chart.session.update [chart] - Update a chart session.
- chart.session.remove [chart] - Remove a chart session.
- chart.session.watch.set [chart] - Enable or disable chart session watch.
- chart.session.assign_tab [chart] - Assign a browser tab to a chart session timeframe.
- chart.session.clear_timeframe [chart] - Clear a chart session timeframe assignment.
- chart.focus [chart] - Focus native chart on symbol/timeframe.
- chart.symbol.set [chart] - Set native chart symbol.
- chart.timeframe.set [chart] - Set native chart timeframe.
- chart.overlay.setups.toggle [chart] - Toggle setup overlays on native chart.
- chart.overlay.reviews.toggle [chart] - Toggle review overlays on native chart.
- chart.show.indicators.set [chart] - Show/hide indicators on native chart.
- chart.show.live_quote.set [chart] - Show/hide live quote on native chart.
- chart.show.ranges.set [chart] - Show/hide range overlays on native chart.
- chart.show.sessions.set [chart] - Show/hide session overlays on native chart.
- chart.show.constraints.set [chart] - Show/hide broker constraints on native chart.
- chart.show.positions.set [chart] - Show/hide positions on native chart.
- chart.show.orders.set [chart] - Show/hide orders on native chart.
- chart.frame.toggle [chart] - Toggle a chart timeframe on native chart.
- chart.frames.set [chart] - Set active chart timeframes on native chart.
- chart.refresh [chart] - Refresh native chart data.
- chart.capture_all [chart] - Capture multi-frame native chart snapshot.
- chart.capture_snapshot [chart] - Capture a native chart snapshot.
- chart.fullscreen.toggle [chart] - Toggle native chart fullscreen mode.
- playbook.list [playbook] - List available task playbooks.
- playbook.save [playbook] - Create or update a task playbook.
- playbook.run [playbook] - Run a multi-step task playbook.
- playbook.run.list [playbook] - List playbook runs from the ledger.

- playbook.run.get [playbook] - Fetch a single playbook run by id.
- playbook.run.resume [playbook] - Resume or approve a blocked playbook run.
- backtest.summary [backtest] - Fetch backtest summary from panel.
- backtest.training_pack [backtest] - Fetch backtest training pack.
- backtest.run [backtest] - Run a single backtest with explicit params.
- backtest.export [backtest] - Export a backtest run or optimization result.
- backtest.optimization [backtest] - Run headless backtest optimization.
- backtest.optimization.cancel [backtest] - Cancel a running headless backtest optimization.
- backtest.optimizer.start [backtest] - Start multi-objective backtest optimizer.
- backtest.optimizer.status [backtest] - Check optimizer status.
- backtest.optimizer.results [backtest] - Fetch optimizer results.
- backtest.optimizer.refine [backtest] - Propose optimizer refinement.
- backtest.optimizer.chain [backtest] - Run chained optimization (round 1 + round 2).
- backtest.optimizer.winner_params [backtest] - Fetch optimizer winner parameters.
- backtest.optimizer.save_preset [backtest] - Save optimizer winner to setup library.
- backtest.apply_optimization [backtest] - Apply optimized parameters to the backtester panel.
- backtester.execution.set [backtest] - Update Backtester execution config.
- backtester.confluence.set [backtest] - Update Backtester confluence config.
- backtester.validation.set [backtest] - Update Backtester validation config.
- backtester.walkforward.set [backtest] - Update Backtester walk-forward config.
- backtester.replay.set [backtest] - Update Backtester replay controls.
- backtester.replay.play.set [backtest] - Toggle Backtester replay playback.
- backtester.replay.step [backtest] - Step Backtester replay forward/backward.
- backtester.tiebreaker.set [backtest] - Update Backtester tie-breaker preference.
- backtester.auto_summary.set [backtest] - Update Backtester auto-summary settings.
- backtester.watchlist.mode.set [backtest] - Set Backtester watchlist mode (suggest/paper/live).
- backtester.trade.select [backtest] - Select a trade in Backtester results.
- backtester.memory.filters.set [backtest] - Update Backtester agent memory filters.
- backtester.research.config.set [backtest] - Update Backtester research autopilot config.
- backtest.preset.list [backtest] - List backtester optimizer presets.
- backtest.preset.save [backtest] - Save or update a backtester optimizer preset.
- backtest.preset.load [backtest] - Load a backtester optimizer preset.
- backtest.preset.delete [backtest] - Delete a backtester optimizer preset.
- backtest.preset.export [backtest] - Export backtester optimizer presets.
- backtest.preset.import [backtest] - Import backtester optimizer presets from JSON.
- backtest.batch.preset.list [backtest] - List backtester batch presets.
- backtest.batch.preset.save [backtest] - Save or update a backtester batch preset.
- backtest.batch.preset.load [backtest] - Load a backtester batch preset.
- backtest.batch.preset.delete [backtest] - Delete a backtester batch preset.

- backtest.batch.run [backtest] - Run backtester batch optimization.
- backtest.batch.cancel [backtest] - Cancel a running backtester batch optimization.
- backtest.batch.clear [backtest] - Clear backtester batch results.
- backtest.batch.export [backtest] - Export backtester batch results.
- research.autopilot.start [backtest] - Start research autopilot session.
- research.autopilot.status [backtest] - Check research autopilot status.
- research.autopilot.results [backtest] - Fetch research autopilot results.
- research.autopilot.stop [backtest] - Stop research autopilot session.
- broker.refresh_snapshot [broker] - Refresh broker snapshot.
- broker.close_position [broker] - Close an open broker position.
- broker.cancel_order [broker] - Cancel an open broker order.
- broker.modify_position [broker] - Modify an open position SL/TP.
- broker.modify_order [broker] - Modify an open order price/qty/SL/TP.
- broker.quote [broker] - Fetch broker quote.
- watcher.list [watcher] - List setup watchers.
- watcher.create [watcher] - Create a setup watcher.
- watcher.update [watcher] - Update a setup watcher.
- watcher.delete [watcher] - Delete a setup watcher.
- watcher.signals [watcher] - Fetch setup signals.
- watcher.signal.explain [watcher] - Explain setup signal.
- watcher.bulk.enable [watcher] - Enable multiple setup watchers.
- watcher.bulk.disable [watcher] - Disable multiple setup watchers.
- watcher.bulk.mode [watcher] - Set mode for multiple setup watchers.
- setup.library.list [setup] - List setup library entries.
- setup.library.create_watcher [setup] - Create watcher from setup library.
- setup.library.save [setup] - Save a setup library entry from explicit params.
- setup.library.update [setup] - Update a setup library entry from explicit params.
- setup.library.delete [setup] - Delete a setup library entry.
- setup.library.export [setup] - Export setup library entries.
- setup.library.import [setup] - Import setup library entries from JSON.
- setups.filters.set [setup] - Set Setups panel filters.
- setups.signals.clear [setup] - Clear setup signals list.
- setups.replay.range.set [setup] - Set setup replay range days.
- agent.memory.get [agent] - Fetch a stored agent memory record.
- agent.memory.clear [agent] - Clear all agent memory records.
- agent.memory.filters.set [agent] - Update Agent Memory panel filters.
- agent.memory.list [agent] - List agent memory records.
- agent.memory.export [agent] - Export agent memory records.
- agent_test.scenario.create [agent_test] - Create or update an agent test scenario.

- agent_test.scenario.list [agent_test] - List agent test scenarios.
- agent_test.scenario.get [agent_test] - Fetch a single agent test scenario.
- agent_test.run [agent_test] - Run an agent test scenario using a playbook.
- agent_test.run.list [agent_test] - List agent test runs.
- agent_test.run.get [agent_test] - Fetch a single agent test run.
- agent.capabilities.update [agent] - Update agent capability flags.
- agent.add [agent] - Add a new chat agent.
- agent.update [agent] - Update a chat agent profile.
- agent.delete [agent] - Delete a chat agent profile.
- agent.switch [agent] - Switch active chat agent.
- autopilot.update_settings [autopilot] - Update AutoPilot risk settings.
- autopilot.enable [autopilot] - Enable AutoPilot.
- autopilot.disable [autopilot] - Disable AutoPilot.
- autopilot.mode.set [autopilot] - Set AutoPilot mode.
- autopilot.execution_mode.set [autopilot] - Set AutoPilot execution mode (live/paper/shadow).
- autopilot.confirmation.set [autopilot] - Toggle AutoPilot confirmation requirement.
- autopilot.killswitch.set [autopilot] - Toggle AutoPilot kill switch.
- autopilot.memory.filters.set [autopilot] - Update AutoPilot memory manager filters.
- autopilot.flow.filters.set [autopilot] - Update AutoPilot recommended flow filters.
- autopilot.run.select [autopilot] - Select a Task Tree run in AutoPilot.
- autopilot.telegram.inputs.set [autopilot] - Update AutoPilot Telegram input fields.
- shadow.stats.get [autopilot] - Fetch shadow mode trade stats and compare summary.
- shadow.trades.list [autopilot] - List shadow mode trades from the ledger.
- shadow.compare [autopilot] - Compare shadow trades to live/paper outcomes.
- tradelocker.search_instruments [broker] - Search TradeLocker instruments.
- tradelocker.positions.list [broker] - List TradeLocker open positions (from snapshot).
- tradelocker.orders.list [broker] - List TradeLocker open orders (from snapshot).
- tradelocker.history.list [broker] - List TradeLocker trade history from ledger.
- tradelocker.cancel_all_orders [broker] - Cancel all open TradeLocker orders.
- tradelocker.close_all_positions [broker] - Close all open TradeLocker positions.
- tradelocker.place_order [broker] - Place a TradeLocker order (ticket).
- tradelocker.set_active_account [broker] - Set active TradeLocker account.
- tradelocker.connect [broker] - Connect to TradeLocker.
- tradelocker.config.update [broker] - Update TradeLocker saved configuration (non-secret).
- tradelocker.trading_options.set [broker] - Update TradeLocker trading options.
- tradelocker.stream.start [broker] - Start TradeLocker streaming.
- tradelocker.stream.stop [broker] - Stop TradeLocker streaming.
- tradelocker.disconnect [broker] - Disconnect from TradeLocker.
- tradelocker.refresh_accounts [broker] - Refresh TradeLocker account list.

- tradelocker.clear_secrets [broker] - Clear saved TradeLocker secrets.
- agent.memory.add [agent] - Add a manual trade memory.
- agent.memory.update [agent] - Update a trade memory entry.
- agent.memory.delete [agent] - Delete a trade memory entry.
- notes.append [notes] - Append a note entry to Notes.
- notes.entry.open [notes] - Open a specific note entry by id.
- notes.checklist.toggle [notes] - Toggle a checklist item on the active note.
- notes.mistake.toggle [notes] - Toggle a mistake tag on the active note.
- notes.trade_link.add [notes] - Link a ledger trade to the active note.
- notes.trade_link.remove [notes] - Remove a linked ledger trade from the active note.
- notes.context.attach [notes] - Attach context pack to the active note.
- notes.context.clear [notes] - Clear attached context from the active note.
- notes.export [notes] - Export notes to CSV or Markdown.
- notes.list [notes] - List notes entries.
- notes.filters.set [notes] - Set Notes panel filters.
- notes.auto_link.set [notes] - Toggle auto-link for new notes.
- notes.auto_recap.set [notes] - Toggle weekly recap auto-generation.
- notes.calendar.set_month [notes] - Set Notes calendar month.
- notes.update [notes] - Update a notes entry.
- notes.delete [notes] - Delete a notes entry.
- audit.list [ledger] - List audit events from the ledger.
- audit.filters.set [ledger] - Update audit trail panel filters.
- audit.export [ledger] - Export audit events.
- truth.replay [ledger] - Replay truth events for a run or symbol.
- task_tree.replay [ledger] - Replay a task tree run by id.
- task_tree.resume [task_tree] - Resume a blocked task tree run after restart.
- task_tree.abort [task_tree] - Abort a pending task tree run.
- changes.list [ledger] - List change events (audit trail).
- changes.export [ledger] - Export change events (audit trail).
- changes.filters.set [ui] - Update Changes panel filters (symbol/range/limit).
- ledger.stats [ledger] - Fetch trade ledger stats.
- dashboard.session.select [ui] - Select a dashboard session and filters.
- dashboard.refresh [ui] - Refresh the performance dashboard data.
- dashboard.action.apply [ui] - Apply a dashboard experiment/champion.
- dashboard.action.promote [ui] - Promote a dashboard experiment/champion to a watch profile.
- dashboard.action.save_preset [ui] - Save a dashboard experiment/champion to preset library.
- dashboard.autopilot.target [ui] - Start a targeted Autopilot session for a regime key.
- mt5.connect [mt5] - Connect to the MT5 tick bridge.
- mt5.disconnect [mt5] - Disconnect from the MT5 tick bridge.

- mt5.ws_url.set [mt5] - Set MT5 bridge WebSocket URL.
- mt5.symbols.set [mt5] - Set MT5 subscribed symbols list.
- mt5.symbols.apply [mt5] - Apply MT5 symbol subscriptions.
- mt5.symbols.search [mt5] - Search MT5 broker symbols.
- mt5.log.open [mt5] - Open MT5 bridge log file.
- trade.propose [broker] - Propose a trade setup.